William Stallings
Data and Computer Communications
7th Edition

Chapter 22
Distributed Applications

Electronic Mail

- Most heavily used application on any network
- Simple Mail Transfer Protocol (SMTP)
  - —TCP/IP
  - —Delivery of simple text messages
- Multi-purpose Internet Mail Extension (MIME)
  - —Delivery of other types of data
  - —Voice, images, video clips

SMTP

- RFC 821
- Not concerned with format of messages or data
  - Covered in RFC 822 (see later)
- SMTP uses info written on envelope of mail
  - Message header
- Does not look at contents
  - Message body
- Except:
  - Standardize message character set to 7 bit ASCII
  - Add log info to start of message
    - Shows path taken

Basic Operation

- Mail created by user agent program (mail client)
  - —Message consists of:
    - Header containing recipient's address and other info
    - Body containing user data
- Messages queued and sent as input to SMTP sender program
  - —Typically a server process (daemon on UNIX)

Mail Message Contents

- Each queued message has:
  - —Message text
    - RFC 822 header with message envelope and list of recipients
    - Message body, composed by user
  - —A list of mail destinations
    - Derived by user agent from header
    - May be listed in header
    - May require expansion of mailing lists
    - May need replacement of mnemonic names with mailbox names
- If BCCs indicated, user agent needs to prepare correct message format

SMTP Sender

- Takes message from queue
- Transmits to proper destination host
  - Via SMTP transaction
  - Over one or more TCP connections to port 25
- Host may have multiple senders active
- Host should be able to create receivers on demand
- When delivery complete, sender deletes destination from list for that message
- When all destinations processed, message is deleted

Optimization

- If message destined for multiple users on a given host, it is sent only once
  - —Delivery to users handled at destination host
- If multiple messages ready for given host, a single TCP connection can be used
  - —Saves overhead of setting up and dropping connection

Possible Errors

- Host unreachable
- Host out of operation
- TCP connection fail during transfer
- Sender can re-queue mail
  - —Give up after a period
- Faulty destination address
  - —User error
  - —Target user changed address
  - —Redirect if possible
  - —Inform user if not

SMTP Protocol - Reliability

- Used to transfer messages from sender to receiver over TCP connection
- Attempts to provide reliable service
- No guarantee to recover lost messages
- No end to end acknowledgement to originator
- Error indication delivery not guaranteed
- Generally considered reliable

SMTP Receiver

- Accepts arriving message
- Places in user mailbox or copies to outgoing queue for forwarding
- Receiver must:
  - —Verify local mail destinations
  - —Deal with errors
    - Transmission
    - Lack of disk space
- Sender responsible for message until receiver confirm complete transfer
  - —Indicates mail has arrived at host, not user

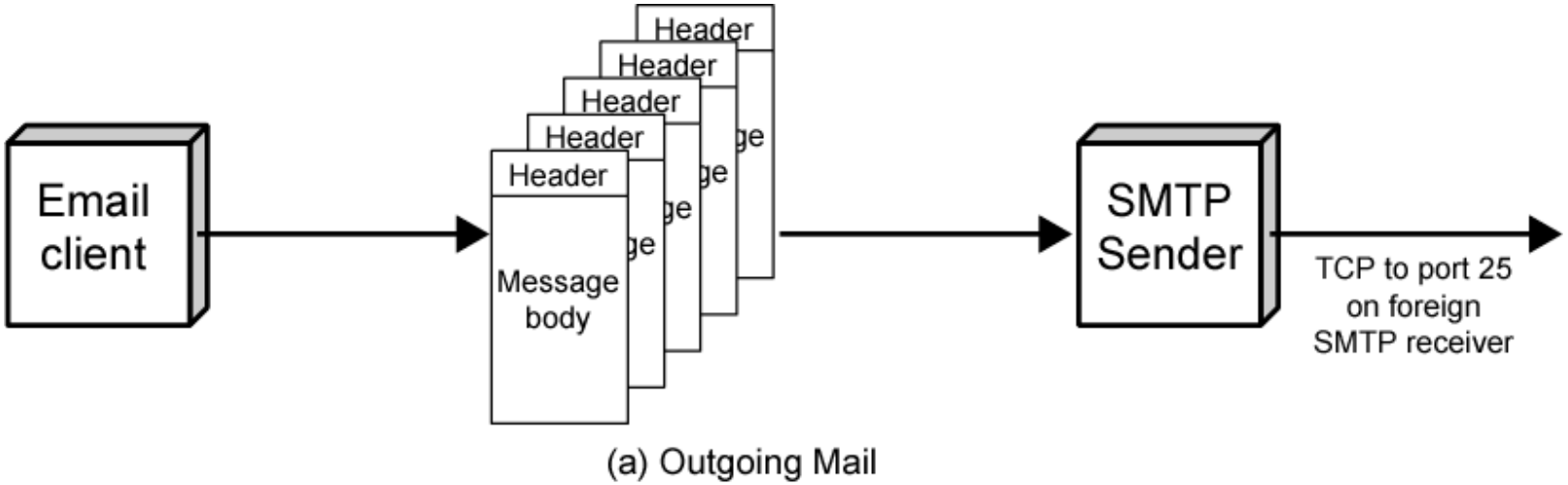
SMTP Forwarding

- Mostly direct transfer from sender host to receiver host
- May go through intermediate machine via forwarding capability
  - —Sender can specify route
  - —Target user may have moved

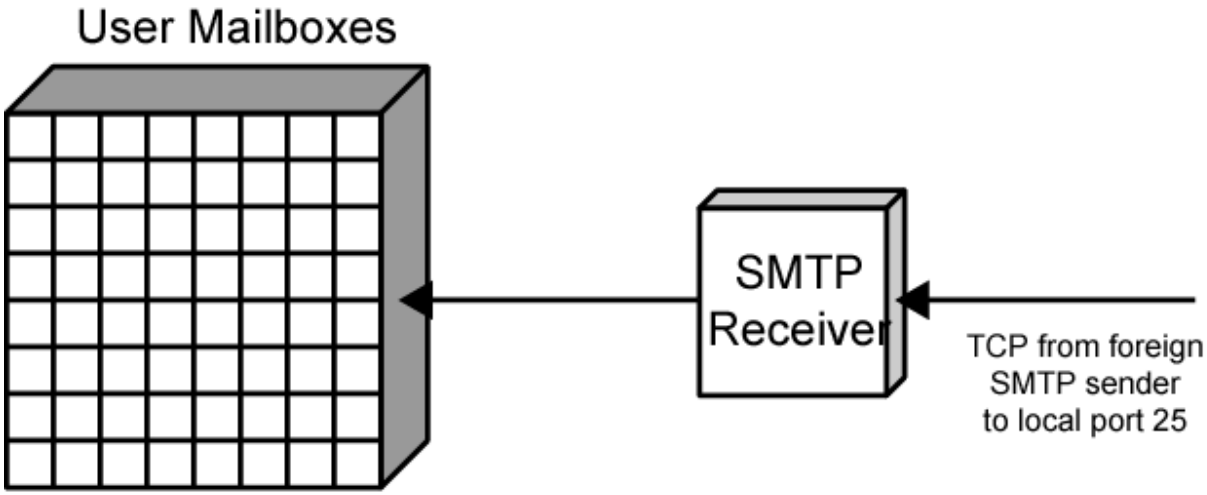
Conversation

- SMTP limited to conversation between sender and receiver
- Main function is to transfer messages
- Rest of mail handling beyond scope of SMTP
  - —May differ between systems

SMTP Mail Flow



(a) Outgoing Mail



(b) Incoming Mail

SMTP System Overview

- Commands and responses between sender and receiver
- Initiative with sender
  - —Establishes TCP connection
- Sender sends commands to receiver
- e.g. HELO<SP><domain><CRLF>
- Each command generates exactly one reply
- e.g. 250 requested mail action ok; completed

SMTP Replies

- Leading digit indicates category
  - —Positive completion reply (2xx)
  - —Positive intermediate reply (3xx)
  - —Transient negative completion reply (4xx)
  - —Permanent negative completion reply (5xx)

Operation Phases

- Connection setup
- Exchange of command-response pairs
- Connection termination

Connection Setup

- Sender opens TCP connection with receiver
- Once connected, receiver identifies itself
  - —220 <domain> service ready
- Sender identifies itself
  - —HELO
- Receiver accepts sender's identification
  - —250 OK
- If mail service not available, step 2 above becomes:
  - —421 service not available

Mail Transfer

- Sender may send one or more messages to receiver
- MAIL command identifies originator
  - Gives reverse path to used for error reporting
  - Receiver returns 250 OK or appropriate fail/error message
- One or more RCPT commands identifies recipients for the message
  - Separate reply for each recipient
- DATA command transfers message text
  - End of message indicated by line containing just period (.)

Closing Connection

- Two steps
- Sender sends QUIT and waits for reply
- Then initiate TCP close operation
- Receiver initiates TCP close after sending reply to QUIT

Format for Text Messages RFC 882

- Message viewed as having envelope and contents
- Envelope contains information required to transmit and deliver message
- Message is sequence of lines of text
  - —Uses general memo framework
  - —Header usually keyword followed by colon followed by arguments

Example Message

Date:Tue, 16 Jan 1996 10:37:17 (EST)

From: “William Stallings” <ws@host.com>

Subject:The syntax of RFC 822

To: Smith@otherhost.com

Cc: Jones@Yet-another_host.com

This is the main text, delimited from the header by a blank line.

Multipurpose Internet Mail Extension (MIME)

- Extension to RFC822
- SMTP can not transmit executables
  - Uuencode and other schemes are available
    - Not standardized
- Can not transmit text including international characters (e.g. â, å, ä, è, é, ê, ë)
  - Need 8 bit ASCII
- Servers may reject mail over certain size
- Translation between ASCII and EBCDIC not standard
- SMTP gateways to X.400 can not handle none text data in X.400 messages
- Some SMTP implementations do not adhere to standard
  - CRLF, truncate or wrap long lines, removal of white space, etc.

Overview of MIME

- Five new message header fields
  - —MIME version
  - —Content type
  - —Content transfer encoding
  - —Content Id
  - —Content Description
- Number of content formats defines
- Transfer encoding defined

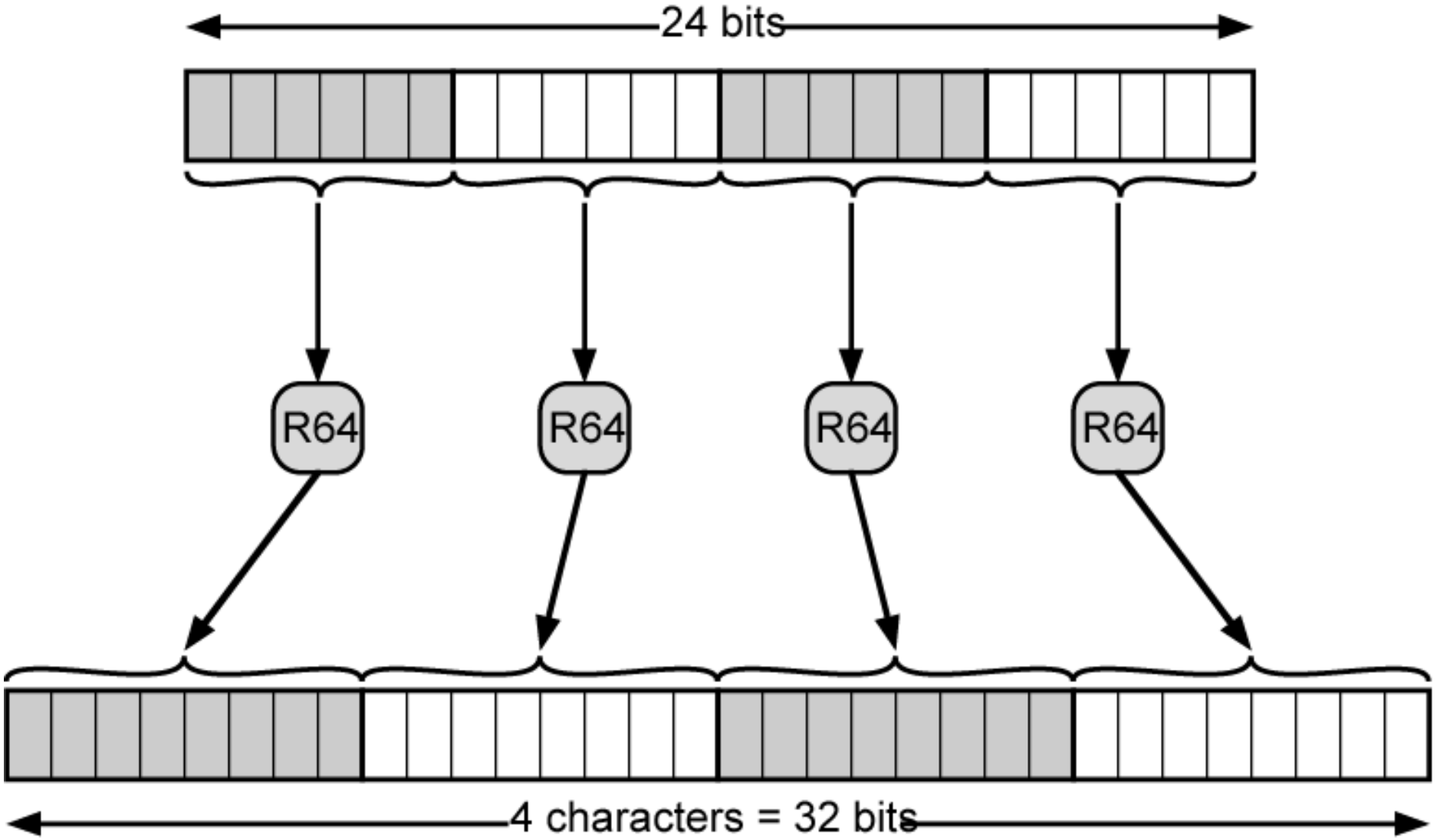
Content Types

- Text body
- Multipart
  - Mixed, Parallel, Alternative, Digest
- Message
  - RFC 822, Partial, External-body
- Image
  - jpeg, gif
- Video
  - mpeg
- Audio
  - Basic
- Application
  - Postscript
  - octet stream

MIME Transfer Encodings

- Reliable delivery across wide largest range of environments
- Content transfer encoding field
  - Six values
  - Three (7bit, 8bit, binary) no encoding done
    - Provide info about nature of data
- Quoted-printable
  - Data largely printable ASCII characters
  - Non-printing characters represented by hex code
- Base64
  - Maps arbitrary binary input onto printable output
- X-token
  - Named nonstandard encoding

Radix-64 Encoding



Hypertext Transfer Protocol HTTP

- Underlying protocol of the World Wide Web
- Not a protocol for transferring hypertext
  - For transmitting information with efficiency necessary for hypertext jumps
- Can transfer plain text, hypertext, audio, images, and Internet accessible information

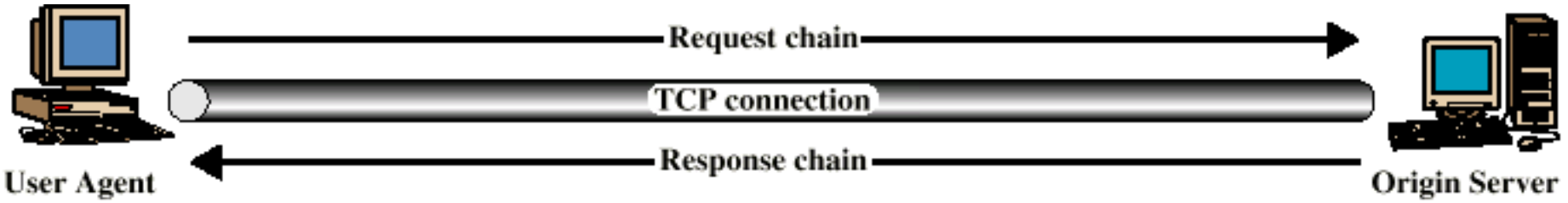
HTTP Overview

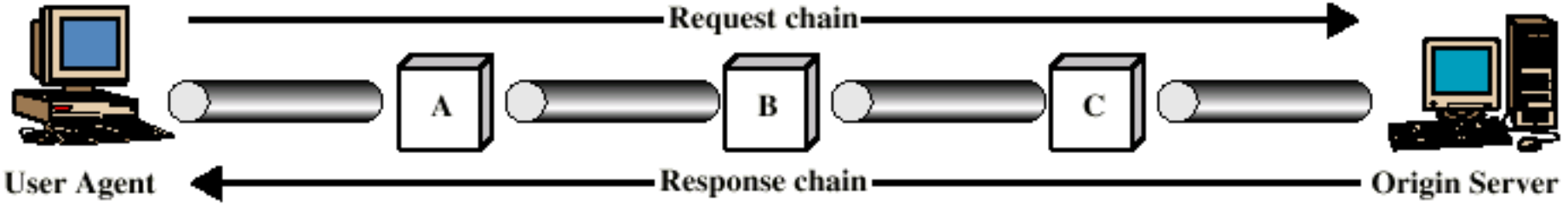
- Transaction oriented client/server protocol
- Usually between Web browser (clinet) and Web server
- Uses TCP connections
- Stateless
  - —Each transaction treated independently
  - —Each new TCP connection for each transaction
  - —Terminate connection when transaction complete

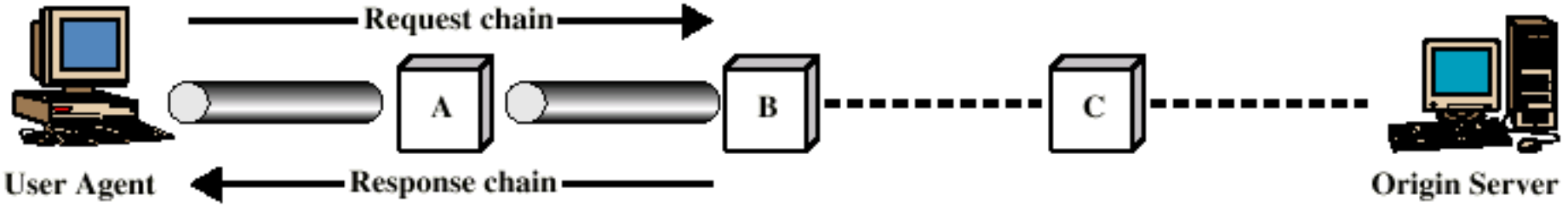
Key Terms

- Cache
- Client
- Connection
- Entity
- Gateway
- Message
- Origin server
- Proxy
- Resource
- Server
- Tunnel
- User agent

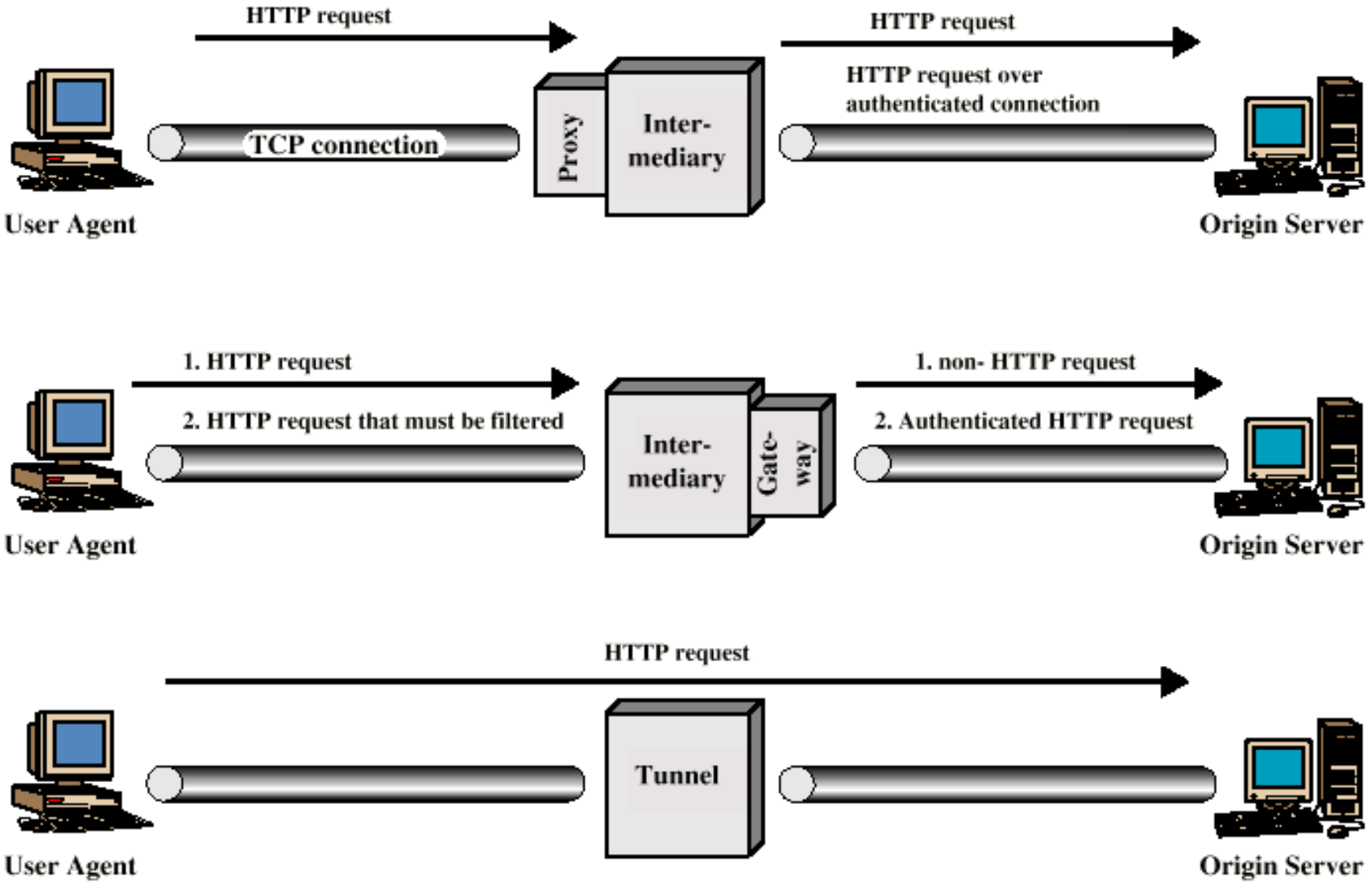
Examples of HTTP Operation







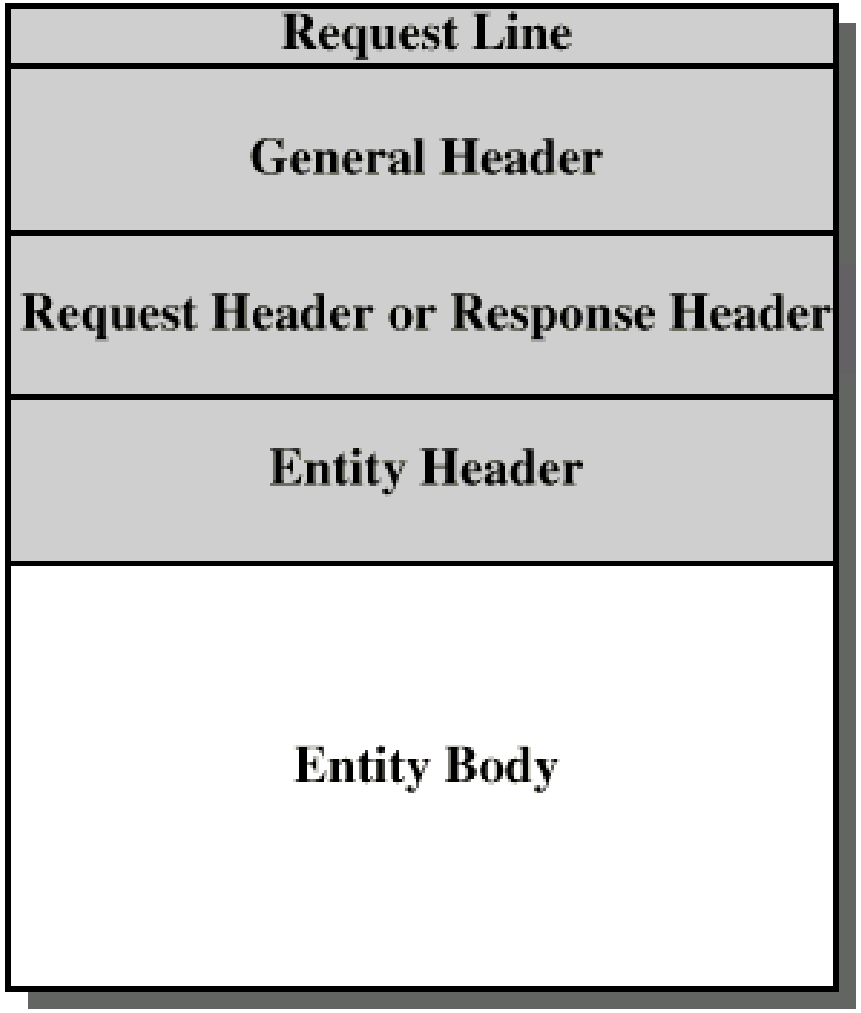
Intermediate HTTP Systems



HTTP Messages

- Requests
  - Client to server
- Responses
  - Server to client
- Request line
- Response line
- General header
- Request header
- Response header
- Entity header
- Entity body

HTTP Message Structure



General Header Fields

- Cache control
- Connection
- Data
- Forwarded
- Keep alive
- MIME version
- Pragma
- Upgrade

Request Methods

- Request-Line = Method <SP> Request_URL <SP> HTTP-Version <CRLF>
- Methods:
  - Options
  - Get
  - Head
  - Post
  - Put
  - Patch
  - Copy
  - Move
  - Delete
  - Link
  - Unlink
  - Trace
  - Wrapped
  - Extension-method

Request Header Field

- Accept
- Accept charset
- Accept encoding
- Accept language
- Authorization
- From
- Host
- If modified since
- Proxy authentication
- Range
- Referrer
- Unless
- User agent

Response Messages

- Status line followed by one or more general, response and entity headers, followed by optional entity body

- Status-Line = HTTP-Version <SP> Status-Code <SP> Reason-Phrase <CRLF>

Status Codes

- Informational
- Successful
- Redirection
- Client error
- Server error

Response Header Fields

- Location
- Proxy authentication
- Public
- Retry after
- Server
- WWW-Authenticate

Entity Header Fields

- Allow
- Content encoding
- Content language
- Content length
- Content MD5
- Content range
- Content type
- Content version
- Derived from
- Expires
- Last modified
- Link
- Title
- Transfer encoding
- URL header
- Extension header

Entity Body

- Arbitrary sequence of octets
- HTTP transfers any type of data including:
  - —text
  - —binary data
  - —audio
  - —images
  - —video
- Interpretation of data determined by header fields
  - —Content encoding, content type, transfer encoding

Network Management - SNMP

- Simple Network Management Protocol
- Networks are becoming indispensable
- More complexity makes failure more likely
- Require automatic network management tools
- Standards required to allow multi-vendor networks
- Covering:
  - Services
  - Protocols
  - Management information base (MIB)

Network Management Systems

- Collection of tools for network management
- Single operator interface
- Powerful, user friendly command set
- Performing most or all management tasks
- Minimal amount of separate equipment
  - i.e. use existing equipment
- View entire network as unified architecture
- Active elements provide regular feedback

Key Elements

- Management station or manager
- Agent
- Management information base
- Network management protocol

Management Station

- Stand alone system or part of shared system
- Interface for human network manager
- Set of management applications
  - —Data analysis
  - —Fault recovery
- Interface to monitor and control network
- Translate manager's requirements into monitoring and control of remote elements
- Data base of network management information extracted from managed entities

Management Agent

- Hosts, bridges, hubs, routers equipped with agent software
- Allow them to be managed from management station
- Respond to requests for information
- Respond to requests for action
- Asynchronously supply unsolicited information

Management Information Base

- MIB
- Representation of network resources as objects
- Each object a variable representing one aspect of managed object
- MIB is collection of access points at agent for management of station
- Objects standardized across class of system
  - —Bridge, router etc.

Network Management Protocol

- Link between management station and agent
- TCP/IP uses SNMP
- OSI uses Common Management Information Protocol (CMIP)
- SNMPv2 (enhanced SNMP) for OSI and TCP/IP

Protocol Capabilities

- Get
- Set
- Notify

Management Layout

- May be centralized in simple network
- May be distributed in large, complex network
  - —Multiple management servers
  - —Each manages pool of agents
  - —Management may be delegated to intermediate manager

Example of Distributed Network Management Configuration

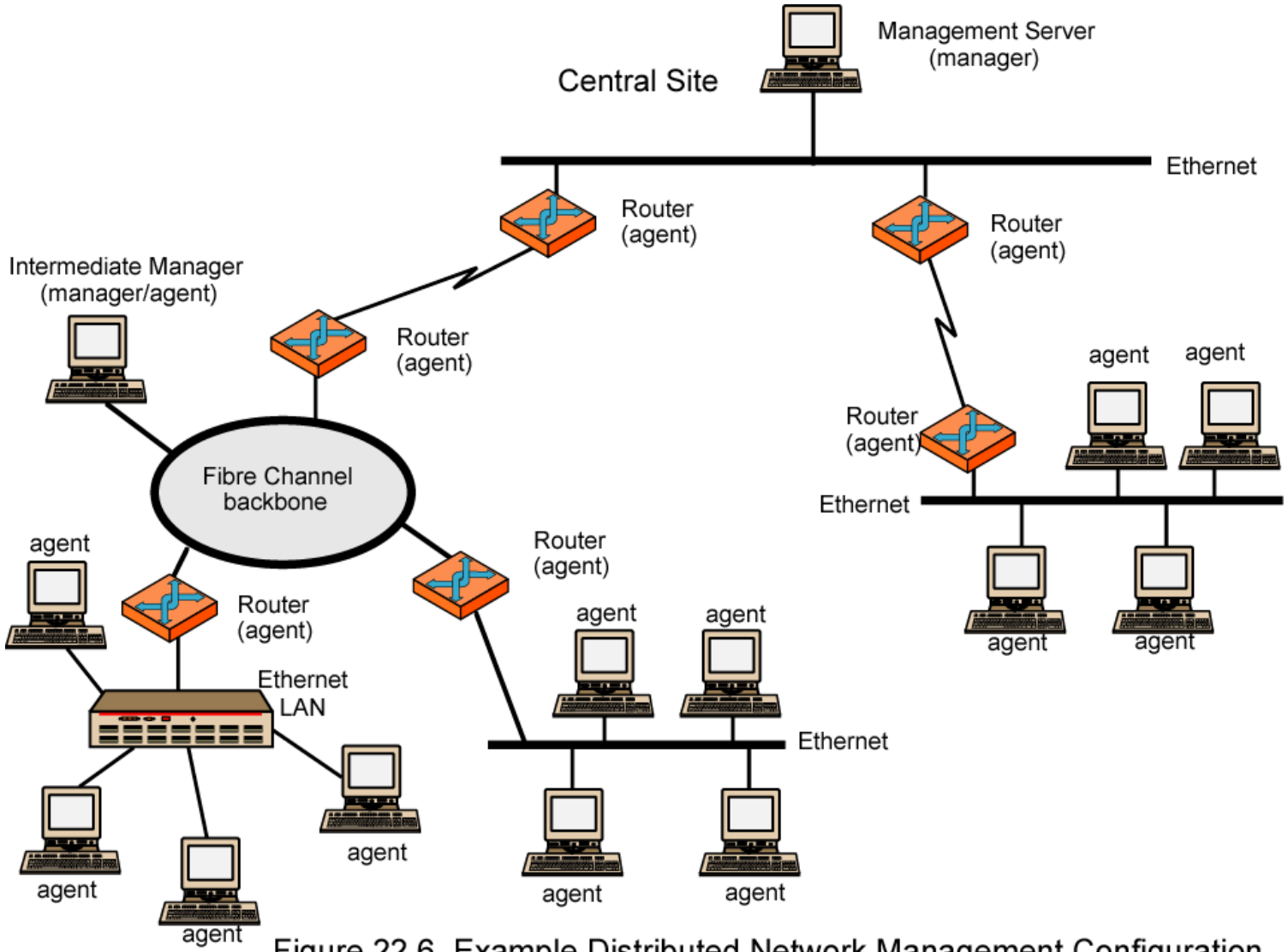
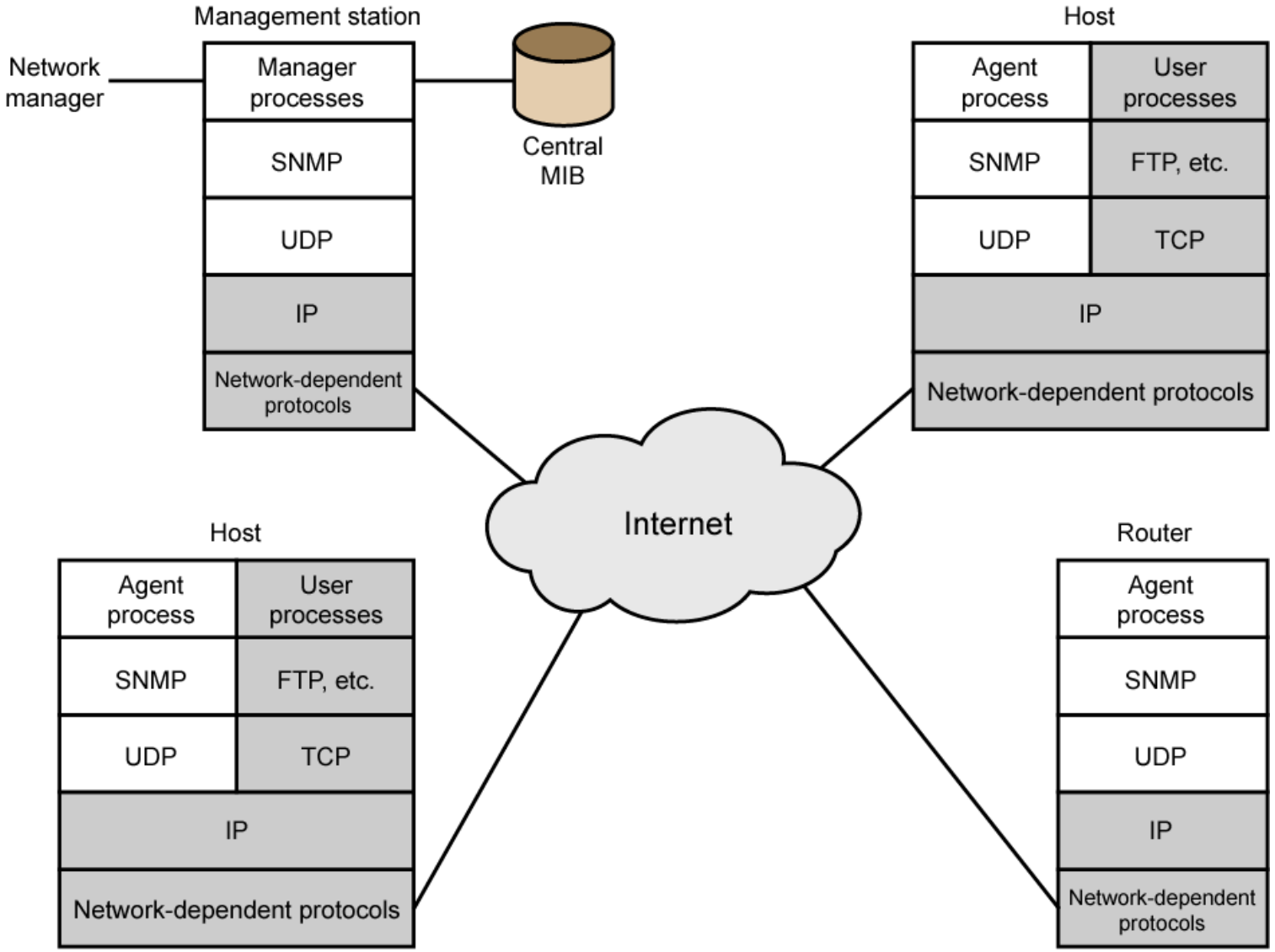


Figure 22.6 Example Distributed Network Management Configuration

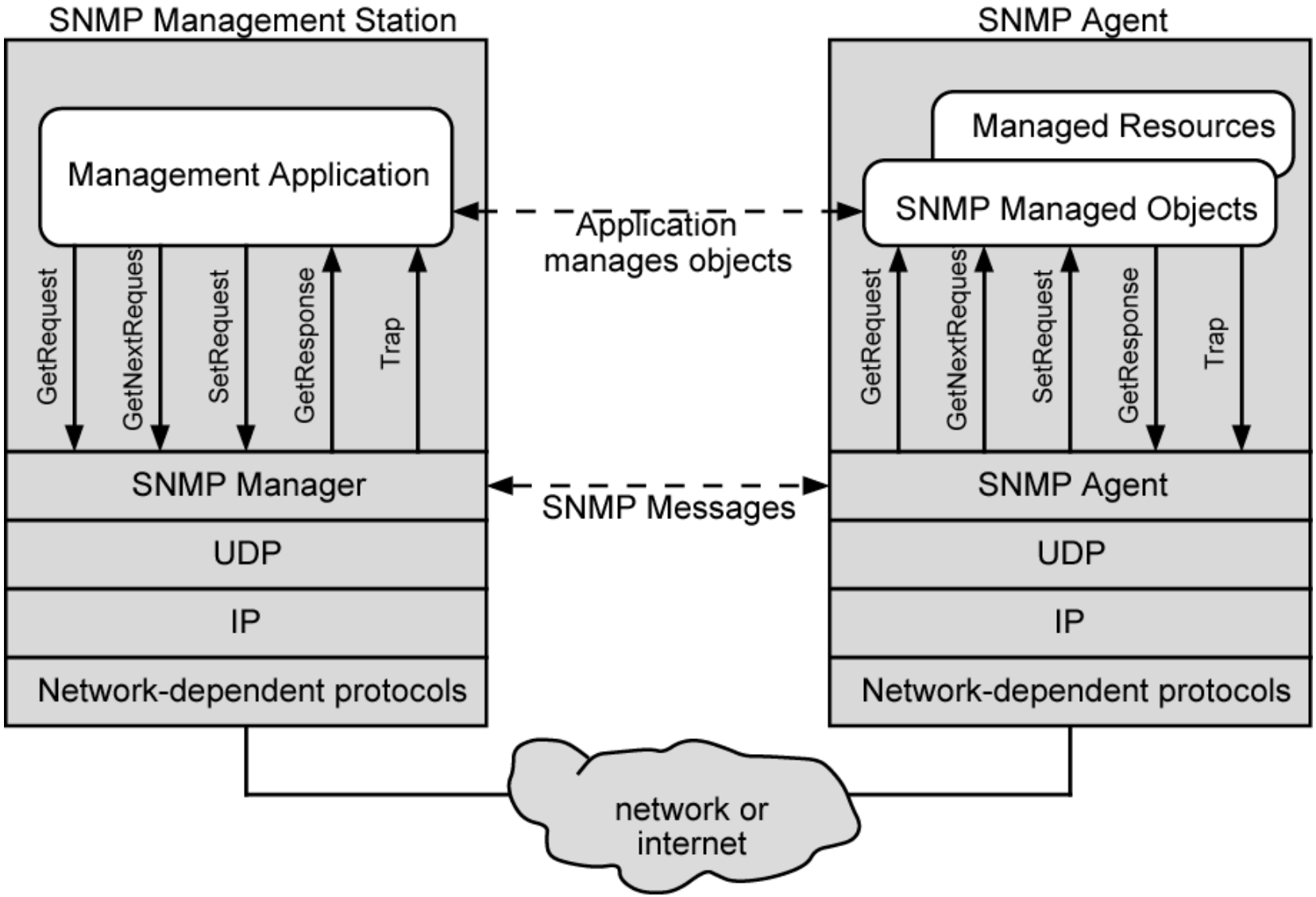
Network Management Protocol Architecture

- Application-level protocol
- Part of TCP/IP protocol suite
- Runs over UDP
- From management station, three types of SNMP messages issued
  - GetRequest, GetNextRequest, and SetRequest
  - Port 161
- Agent replies with GetResponse
- Agent may issue trap message in response to event that affects MIB and underlying managed
  - Port 162

SNMPv1 Configuration



Role of SNMP v1



SNMP v1

- August 1988 SNMP specification issued
- Stand alone management stations and bridges, routers workstations etc supplied with agents
- Defines limited, easily implemented MIB of scalar variables and two dimensional tables
- Streamlined protocol
- Limited functionality
- Lack of security
- SNMP v2 1993, revised 1996
  - RFC 1901-1908

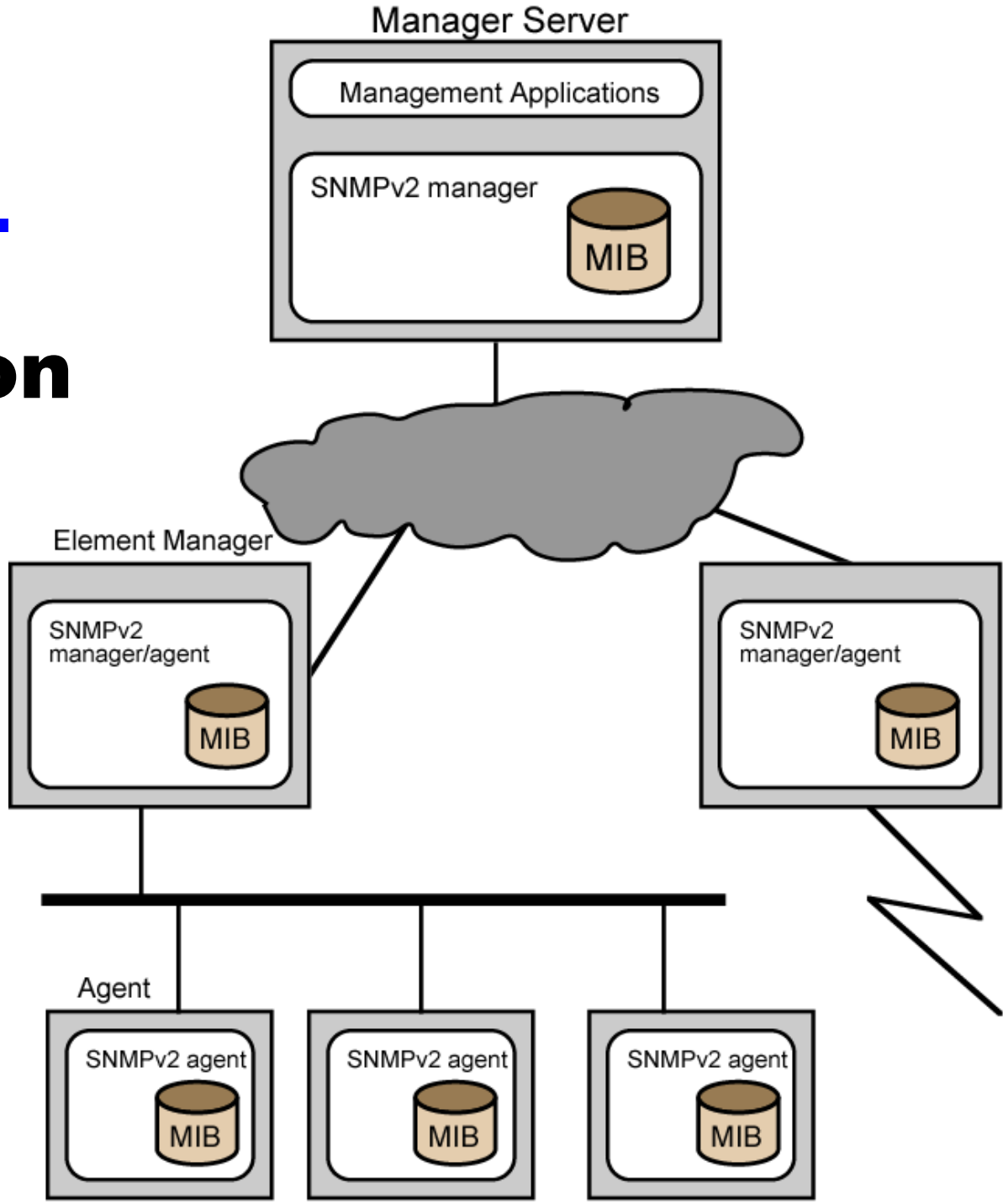
SNMP v2 (1)

- Framework on which network management applications can be built
  - —e.g fault management, performance monitoring, accounting
- Protocol used to exchange management information
- Each player maintains local MIB
  - —Structure defined in standard
- At least one system responsible for management
  - —Houses management applications

SNPM v2 (2)

- Support central or distributed management
- In distributes system, some elements operate as manager and agent
- Exchanges use SNMP v2 protocol
  - —Simple request/response protocol
  - —Typically uses UDP
    - Ongoing reliable connection not required
    - Reduces management overhead

SNMP v2 Managed Configuration



Structure of Management Information

- SMI
- Defines general framework with which MIB defined and constructed
- Identifies data types
- How resources are represented and named
- Encourages simplicity and extensibility
- Scalars and two dimensional arrays of scalars (tables) only

Protocol Operation

- Exchange of messages
- Outer message header deals with security
- Seven types of PDU

SNMP v2 PDU Formats

PDU type	request-id	0	0	variable-bindings

(a) GetRequest-PDU, GetNextRequest-PDU, SetRequest-PDU, SNMPv2-Trap-PDU, InformRequest-PDU

PDU type	request-id	error-status	error-index	variable-bindings

(b) Response-PDU

PDU type	request-id	non-repeaters	max-repetitions	variable-bindings

(c) GetBulkRequest-PDU

name1	value1	name2	value2	• • •	name*n*	value*n*

(d) variable-bindings

SNMP v3

- Addresses security issues of SNMP v1/2
- RFC 2570-2575
- Proposed standard January 1998
- Defines overall architecture and security capability
- To be used with SNMP v2

SNMP v3 Services

- Authentication
  - Part of User-Based Security (UBS)
  - Assures that message:
    - Came from identified source
    - Has not been altered
    - Has not been delayed or replayed
- Privacy
  - Encrypted messages using DES
- Access control
  - Can configure agents to provide a number of levels of access to MIB
  - Access to information
  - Limit operations

Required Reading

- Stallings chapter 22
- WWW Consortium
- Loads of web sites on SNMP